Title	Flow assignment and packet scheduling for multipath networks
Author(s)	Leung, KC; Li, VOK
Citation	The 1999 IEEE Global Telecommunications Conference, Rio de Janeireo, Brazil, 5-9 December 1999. In Globecom. IEEE Global Telecommunications Conference & Exhibition Conference Record, 1999, v. 1 pt. A, p. 246-250
Issued Date	1999
URL	http://hdl.handle.net/10722/46159
Rights	©1999 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

FLOW ASSIGNMENT AND PACKET SCHEDULING FOR MULTIPATH NETWORKS

Ka-Cheong Leung and Victor O. K. Li
Department of Electrical and Electronic Engineering
The University of Hong Kong
Pokfulam Road, Hong Kong, China
{kcleung, vli}@eee.hku.hk

Abstract—In this paper, we propose a framework to study how to route packets efficiently in multipath communication networks. Two traffic congestion control techniques, namely, flow assignment and packet scheduling, have been investigated. The flow assignment mechanism defines an optimal splitting of data traffic on multiple disjoint paths. The resequencing delay and the usage of the resequencing buffer can be reduced significantly by properly scheduling the sending order of all packets, say, according to their expected arrival times at the destination. We consider a multiple-node $M/M/1$ tandem network with a delay line as the path model. When end-to-end path delays are all Gaussian distributed, our analytical results show that the techniques are very effective in reducing the average end-to-end path delay, the average packet resequencing delay, and the average resequencing buffer occupancy for various path configurations. These promising results can form a basis for designing future adaptive multipath protocols.

I. INTRODUCTION

The convergence of the computer, communications, entertainment, and consumer electronics industry is driving an explosive growth in multimedia applications [17]. Recent studies show that multimedia traffic exhibits correlation on various time scales [3]. Such long-range dependence property has a considerable unpleasant impact on queueing performance, and is a dominant characteristic for a number of packet traffic engineering problems [8].

Multipath routing has recently been found to be an effective method to alleviate the effects of such traffic correlation [9], [13]. Precisely, *multipath routing* or *spatial traffic dispersion* [6], [7], [9], [20], [21] is a load balancing technique in which the total load from a source to a destination is spatially distributed over several paths. To make the scheme viable, there must be a way to allocate a proper portion of traffic to each participating path so as to satisfy the desired objective, like the minimization of the average end-to-end path delay. This is known as *flow assignment*.

Besides, packets which travel along different paths may arrive out of order. Those packets arriving out of order may have to be resequenced, i.e. stored in a buffer, called a resequencing buffer, until they can be delivered to the end process in the proper order. To provide better performance, the need for resequencing should be minimized [5].

Cyclic traffic dispersion is a method to evenly distribute packets over all active paths. Many existing traffic dispersion strategies, such as dispersity routing [6], [20], [21], the string-mode protocol [7], and the vector routing algorithm [15], utilized this idea. The major advantage of this approach is that it is quite simple to implement. However, this method does not take the heterogeneity of these paths into account. With heterogeneous paths, the best way to spread traffic along multiple paths may not be by cyclic dispersion, since it may not achieve the objective, such as minimizing the end-to-end path delay. In addition, it may induce substantial packet resequencing delay when the end-to-end delays are quite different among these paths.

Some adaptive routing schemes [2], [14] have been proposed to spread packets dynamically over multiple paths according to the network load. These procedures still require some parameters to determine the load distribution. Yet, the calculations of such parameters are either computationally intensive or done in an ad hoc manner. Thus, there is a need for designing new multipath routing schemes that allow a rapid computation of the optimal load distribution parameter.

Since resequencing is due to packets arriving out of order at the destination, instead of sending packets from the source according to their sequence numbers, the packet sending order can be *scheduled* anew to minimize or to reduce the necessity for resequencing.

The objective of this work is to propose a framework to study how to route packets efficiently in multipath communication networks. Two traffic congestion control techniques, namely, flow assignment and packet scheduling, have been investigated. The flow assignment mechanism defines an optimal splitting of data traffic on multiple disjoint paths, whereas the resequencing delay and the usage of the resequencing buffer can be reduced significantly by properly scheduling the sending order of all packets, say, according to their expected arrival times at the destination. We will investigate the effectiveness of these control mechanisms for multipath routing by examining three basic questions:

- What is the optimal split of traffic to achieve the best performance?
- How much does the packet scheduling technique improve the performance?
- Under what circumstances should we employ the optimal split of traffic instead of cyclic traffic dispersion, and the packet scheduling technique, in order to achieve a significant performance improvement?

This paper is organized as follows. Section II gives a flow assignment model to compute an optimal splitting of data traffic on multiple disjoint paths. Section III presents a packet scheduling model to find a proper sending schedule so as to reduce the packet resequencing delay and the resequencing buffer occupancy. Section IV develops an end-to-end path delay model for the framework. Section V examines the analytical results derived from the framework and studies the effectiveness of the flow assignment and packet scheduling techniques on multipath routing. Section VI gives some conclusions and discusses some possible extensions to our work. Lemma and theorem proofs are omitted here because of constraints in space.

II. FLOW ASSIGNMENT MODEL

Our flow model consists of a disordering network connecting the source to the destination, and flows of packets. The disordering network consists of a set of N disjoint paths, namely Path 1, Path 2, ...,

Path N, connecting the source to the destination, such that packets may arrive at the destination in a different order as they are sent. We assume the source has an unlimited supply of packets, which are delivered to paths according to deterministic or probabilistic routing. The relative load to Queue i (or Path i), p_i, is defined as the portion of dispersed traffic to be routed to Queue i, where $\sum_{i=1}^{N} p_i = 1$. Denote the flow configuration $p = (p_1 \quad p_2 \quad \ldots \quad p_N)$. Because of limited path capacity, it may not be possible to route all packets to some of these paths. Therefore, the load distribution p_i is feasible if and only if $0 \le p_i < M_i \le 1$, where M_i is the upper bound of p_i, for all $i = 1, 2, \ldots, N$. A flow configuration is said to be a *feasible flow configuration* if and only if $0 \le p_i < M_i \le 1$ for all $i = 1, 2, \ldots, N$.

Lemma 1: The set of all feasible flow configurations, $\mathcal{S}$, is convex.

Let $C_i(p_i)$ denote the non-negative monotonically increasing cost function of transmitting packets on Path i with the relative load p_i, where $i = 1, 2, \ldots, N$. $C_i(p_i)$ is well defined for all feasible flow configurations. The notion of 'cost' here is quite general. It can represent any system and user costs, including the average end-to-end path delay, the average packet loss probability, the average total buffer requirement for all nodes along the path, and so on. Hence, the expected cost for multipath routing can be written as:

$$\overline{C}(p) = \sum_{i=1}^{N} p_i \, C_i(p_i)$$

Let $C_i(p_i)$ be a non-negative monotonically increasing convex function in p_i.

Lemma 2: The expected cost $\overline{C}(p)$ is a convex function in p, where p is a feasible flow configuration.

Because the expected cost $\overline{C}(p)$ is a convex function in p and all feasible flow configurations p composes the convex set $\mathcal{S}$, $\overline{C}(\cdot)$ has an unique global minimum on $\mathcal{S}$ if $\mathcal{S}$ is non-empty [19]. Iterative feasible direction algorithms like the constrained gradient projection method [4] can be adopted to compute the minimum expected cost and the corresponding feasible flow configuration.

To complete our flow assignment model, the cost function $C_i(p_i)$ is formulated as the average end-to-end delay on Path i, $\overline{D}_i(p_i)$, for our succeeding analytical studies. That is,

$$C_i(p_i) = \overline{D}_i(p_i)$$

where $i = 1, 2, \ldots, N$.

III. PACKET SCHEDULING MODEL

In Section II, we have proposed to employ the flow assignment method to minimize the expected cost for multipath routing. However, this method merely gives the load distributions to N disjoint paths. Given a sequence of W ordered packets to be transmitted, there are many ways to order the sending sequence for these packets from the source. Each order of sending sequence for a set of packets is called a *sending schedule*. A sending schedule can have very different packet resequencing delay and resequencing buffer distributions when compared with another schedule. This means that care must be taken to determine the best schedule. We propose a packet scheduling mechanism to minimize or to reduce the consumption of the resequencing buffer and the delay on resequencing.

Suppose p_i is the relative load on Path i, where $\sum_{i=1}^{N} p_i = 1$. Denote $p = (p_1 \quad p_2 \quad \ldots \quad p_N)$. To achieve the best system performance, packets are distributed to each path in a cyclical fashion so that the arrival instants of any two packets to each path is as uniform as possible. Algorithms for constructing such routing sequences can be found in [1]. These routing sequences contain subsequences that can be used to re-construct the whole sequence by the repeated applications of the same subsequence.

A *round* is defined as the minimal subsequence. It is possible to construct a round of length L such that q_i packets are sent on Path i, $i = 1, 2, \ldots, N$, in each round, where $p_i \approx \frac{q_i}{L}$ and $L = \sum_{j=1}^{N} q_j$. Thus, we can denote $\mathcal{P}_i(j) = k$ as the position of the jth packet in every round to be sent on Path i, and $\mathcal{R}(k) = i$. The sequence number of the packet sent at Round r, $r \ge 1$, and Position k is designated as $\mathcal{O}(r, k)$, where $\mathcal{O}(r, k) \ge 1$. Denote by t_m, d_m, and a_m, respectively, the sending time of the first bit of the packet with the sequence number m from the source, the end-to-end path delay experienced by the packet, and the arrival time of the last bit of the packet at the destination. The relationship among these three quantities is:

$$a_m = t_m + d_m$$

Let $R(s)$ denote the resequencing cost function for multipath routing under the sending schedule s. Our objective is to find an optimal sending schedule s^* such that $R(s)$ is minimized at $s = s^*$.

To complete our packet scheduling model, the resequencing cost function $R(s)$ is expressed as:

$$R(s) = \sum_{k=1}^{W} \max_{l=1}^{k-1}(0, \overline{a_l} - \overline{a_k})$$

where $\overline{a_k} = E[a_k]$.

$R(s)$ can be minimized to take on value 0 if it is possible to obtain the optimal sending schedule such that the expected arrival times for all packets are in ascending order of their sequence numbers. Ties are resolved by sending the packet with the smaller sequence number at an earlier sending time. If all packets arrive at their expected arrival times, resequencing is not needed. The task is reduced to searching for such a schedule.

Denote by $\overline{S}(i, r, j)$ and $\overline{A}(i, r, j)$, respectively, the expected sending time of the first bit of the packet at Round r and Position $\mathcal{P}_i(j)$ from the source, and the expected arrival time of the last bit of the packet at the destination. Without loss of generality, the first packet, i.e. at Round 1 and Position 1, is expected to be sent at Time 0. That is,

$$\overline{S}(\mathcal{R}(1), 1, 1) = 0$$

Thus, the packet at Round r and Position $\mathcal{P}_i(j)$ is expected to be transmitted at:

$$\overline{S}(i, r, j) = [(r - 1) \cdot L + \mathcal{P}_i(j) - 1] \cdot \overline{\Delta}$$

where $\overline{\Delta}$ denotes the average inter-packet time, i.e. the mean time between two successive packets to be transmitted from the source.

Suppose the average end-to-end delay for Path i is $\overline{D}_i$. The expected packet arrival time can be expressed as:

$$\overline{A}(i, r, j) = \overline{S}(i, r, j) + \overline{D}_i$$

Packets are scheduled such that a packet with a smaller sequence number $\mathcal{O}(r_1, \mathcal{P}_{i_1}(j_1))$ is expected to arrive at the destination no later than one with a larger sequence number $\mathcal{O}(r_2, \mathcal{P}_{i_2}(j_2))$. This schedule order relationship can be written as:

$$\mathcal{O}(r_1, \mathcal{P}_{i_1}(j_1)) < \mathcal{O}(r_2, \mathcal{P}_{i_2}(j_2))$$

if and only if

$$\overline{A}(i_1, r_1, j_1) < \overline{A}(i_2, r_2, j_2) \quad \text{or} \quad \begin{cases} \overline{A}(i_1, r_1, j_1) = \overline{A}(i_2, r_2, j_2) \\ \overline{S}(i_1, r_1, j_1) < \overline{S}(i_2, r_2, j_2) \end{cases}$$

It is time to introduce the theorem that guides the construction of the optimal schedule.

Theorem 1: The sequence number of the packet sent at Round r, $r = 1, 2, \ldots$, and Position $\mathcal{P}_i(j)$, $i = 1, 2, \ldots, N$, $j = 1, 2, \ldots, q_i$, can be written as:

$$\mathcal{O}(r, \mathcal{P}_i(j)) = \sum_{\substack{u=1 \\ u \neq i}}^{N} \sum_{v=1}^{q_u} \max(0, r - \xi(i, j, u, v)) + (r-1) \cdot q_i + j$$

where

$$\xi(i, j, u, v) = \begin{cases} \frac{(\overline{D_u} - \overline{D_i}) + [\mathcal{P}_u(v) - \mathcal{P}_i(j)] \cdot \overline{\Delta}}{L \cdot \overline{\Delta}} + 1 & \text{if} \quad \frac{\overline{D_u} - \overline{D_i}}{\overline{\Delta}} \equiv \mathcal{P}_i(j) - \mathcal{P}_u(v) \pmod{L} \quad ; \\ \left\lceil \frac{(\overline{D_u} - \overline{D_i}) + [\mathcal{P}_u(v) - \mathcal{P}_i(j)] \cdot \overline{\Delta}}{L \cdot \overline{\Delta}} \right\rceil & \text{otherwise.} \end{cases}$$

Take $\delta(i, j) = \xi(\mathcal{R}(1), 1, i, j)$. By using Theorem 1, it can be shown that:

$$\mathcal{O}(r+1, \mathcal{P}_i(j)) = \mathcal{O}(r, \mathcal{P}_i(j)) + L$$

where $r \geq \max_{\substack{u, v \\ u \neq \mathcal{R}(1)}} (1, \delta(u, v)) - \min_{\substack{x, y \\ x \neq \mathcal{R}(1)}} (1, \delta(x, y)) + 1$.

The determination for the set of $\delta(i, j)$ characterizes the steady state behavior of the sending schedule, and hence the packet resequencing delay and resequencing buffer occupancy distributions.

IV. END-TO-END PATH DELAY MODEL

The analytical framework proposed in this section is applicable to any distributions of end-to-end path delay. To illustrate our model, we use Gaussian distributed end-to-end path delay. Generally, a path consists of several hops. Traffic measurements on the ARPANET[1] indicated that two sites were separated by 5.32 hops, on the average. Moreover, the distribution of round-trip delay appeared to be roughly normally distributed [12]. Modern high speed networks are much more complex in terms of domain architectures and connectivity. Thus, the mean path length between any two sites in a high speed network tends to be higher than the above value.

The central limit theorem [10] suggests that the end-to-end path delay, which is the sum of a large number of hop delays, is approximately normally distributed. The mean and the variance of the end-to-end path delay provide sufficient information to generate an approximate distribution, which can then be utilized to compute the resequencing delay, the total delay, and the buffer occupancy

---

distributions. This approach, which was used to solve the end-to-end percentile-type delay objective allocation problem for networks supporting Switched Multi-megabit Data Service (SMDS), has been shown to provide the best approximation to the reference values [18].
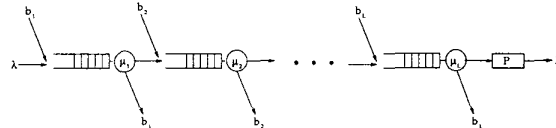

Fig. 1. The L-hop path example.

To complete our end-to-end path delay model, an L-hop path is modeled as a multiple-node $M/M/1$ tandem network with a delay line as illustrated in Figure 1. L $M/M/1$ queues and a delay line are connected in tandem. The ith queue receives input from two traffic sources: the tagged dispersed traffic of rate λ, and the interfering or background traffic of rate b_i. The service rate of the ith server is μ_i. The delay line, which generally represents the total propagation delay of the path, is P. Denote by $\overline{D}$ and σ_D^2 the mean and the variance of the end-to-end path delay. It can be shown [11] that:

$$\overline{D} = \sum_{i=1}^{L} \frac{1}{\mu_i(1 - \rho_i)} + P$$

$$\sigma_D^2 = \sum_{i=1}^{L} \frac{1}{\mu_i^2(1 - \rho_i)^2}$$

where the utilization of the ith server, $\rho_i = \frac{\lambda + b_i}{\mu_i}$.

Since $\frac{\partial \overline{D}}{\partial \lambda} > 0$ and $\frac{\partial^2 \overline{D}}{\partial \lambda^2} > 0$ when λ is evaluated at a value between 0 and $\mu_i - b_i$, $\overline{D}$ is a non-negative monotonically increasing convex function, which can be employed as the cost function to find the unique optimal flow configuration illustrated in Section II.

V. ANALYTICAL INVESTIGATION

This section discusses the numerical results based on the analytical expressions obtained in Sections II, III, and IV. To evaluate the effectiveness of flow assignment and packet scheduling, we need to calculate the resequencing delay and the resequencing buffer occupancy. Those expressions can be found from our proposed resequencing model in [16]. With the help of some numerical examples, we can illustrate the effectiveness of these control mechanisms for multipath routing by answering the three basic questions posed in Section I. The inter-sending time between any two consecutive packets from the source, denoted as the inter-packet time, is five time units. There are two disjoint paths, namely Path 1 and Path 2, connecting the source to the destination. Packets are routed in a cyclical fashion so that the arrival instants of any two packets to each path is as uniform as possible. A path is modeled as a five-node $M/M/1$ tandem network with a fixed delay line. Its end-to-end path delay is assumed to be Gaussian or normally distributed with its mean and variance computed from the path model exhibited in Section IV. Each server on path i, $i = 1, 2$, serves a packet with an average service time of μ_i time units, where $\mu_1 = 1$, and $\mu_2 = 0.5, 1, 2, 5$. The set of fixed delays for Path 1 and Path 2 consists of three different combinations, namely (0, 0), (0, 15), and

---

(a) Load distribution when fixed delays are (0,0) time unit.

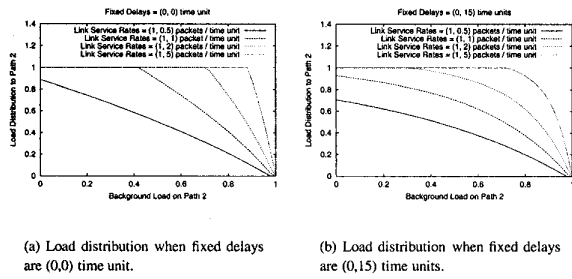(b) Load distribution when fixed delays are (0,15) time units.

Fig. 2. Load distribution plots for various settings.

(15, 0) time units. The background load on Path 1 is fixed to be 0.75 and that on Path 2 varies between 0 and 1, at an increment of 0.02. The results are provided in two sets. The first set studies the effectiveness of the optimal path-splitting of traffic for multipath routing. It provides the load distributions to Path 2, p_2 (where $p_1 = 1 - p_2$), and the average end-to-end path delays. The second set studies the effectiveness of the flow assignment and packet scheduling techniques on resequencing. It includes the average resequencing delays for various settings.

We examine the first set of results. Figure 2 shows the load distributions to Path 2 when the background load on Path 2 varies between 0 and 1. The load distribution drops from 1 to 0 as the background load increases from 0 to 1. Initially, the background load on Path 1 is much higher than that on Path 2. It is beneficial to transmit a majority of packets on Path 2 instead of cyclic dispersion in order to take advantage of a much lower average end-to-end delay on Path 2. Its superiority fades as the background load on Path 2 increases. At a certain background load on Path 2, cyclic dispersion becomes the best strategy for traffic splitting. As the background load on Path 2 continues to rise, it is advantageous to distribute a higher load on Path 1 rather than Path 2. Eventually, all packets should be sent on Path 1.

Besides, it is always possible to achieve optimal splitting with a higher load distribution on Path 2 with faster transmission links. A faster transmission link means that it can take a higher portion of traffic from the source to get the same offered load, given the total work feeding into the network from the source is fixed. Moreover, a higher propagation delay on Path 2 causes a greater end-to-end path delay, resulting in a reduction in the optimal load distribution on Path 2.

Figure 3 exhibits the average end-to-end path delays for both the optimal split of traffic and cyclic dispersion. At low and moderate background load on Path 2, we see that the optimal split of traffic yields a greater reduction on the average delay when the link service rate for Path 2 becomes higher, since an increase in the link service rate reduces the overall offered load and thus the average end-to-end delay on Path 2. In addition, the improvement decreases when the propagation delay on Path 2 is higher, and vice versa.

The second set of results demonstrates the effectiveness of the flow assignment and packet scheduling techniques on resequencing. Our results in Figure 4 show that the optimal flow assignment always yields lower average resequencing delay and lower average resequencing buffer occupancy[2] than cyclic dispersion of traffic.

[2] The average resequencing buffer occupancy is proportional to the average resequencing delay by applying Little's Theorem [11].

(a) Average path delay when fixed delays are (0, 0) time unit and link service rates are (1, 0.5) packets per time unit.

(b) Average path delay when fixed delays are (0, 0) time unit and link service rates are (1, 1) packet per time unit.

(c) Average path delay when fixed delays are (0, 0) time unit and link service rates are (1, 2) packets per time unit.

(d) Average path delay when fixed delays are (0, 0) time unit and link service rates are (1, 5) packets per time unit.

(e) Average path delay when fixed delays are (0, 15) time units and link service rates are (1, 1) packet per time unit.

(f) Average path delay when fixed delays are (15, 0) time units and link service rates are (1, 1) packet per time unit.
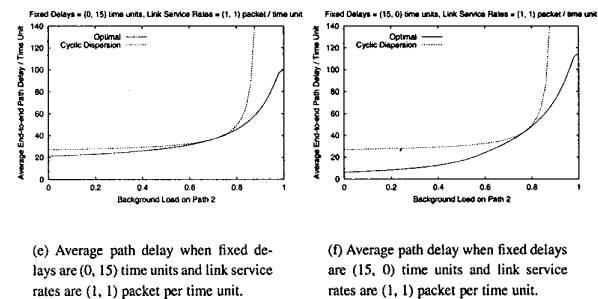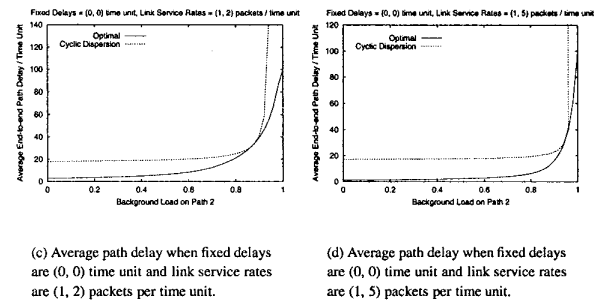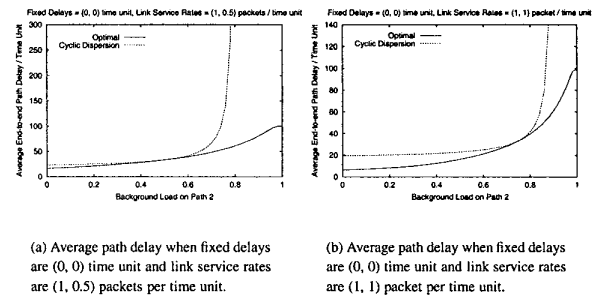
Fig. 3. Average end-to-end path delay plots for various settings.

A further improvement can be obtained by applying the proposed packet scheduling technique. We see that a substantial performance improvement can be achieved for cases where there is a large difference in path delays only when the packet scheduling technique is applied. This means that the packet scheduling mechanism is very effective in providing performance improvement when there is a large difference in path fixed delays.

VI. CONCLUSIONS

In this paper, we propose a framework to study how to route packets efficiently in multipath communication networks. Two traffic congestion control techniques, namely, flow assignment and packet scheduling, have been investigated. The flow assignment mechanism defines an optimal splitting of data traffic on multiple disjoint paths, so as to minimize the average cost for multipath routing, such as the average end-to-end path delay experienced by a

(a) Average resequencing delay when fixed delays are (0, 0) time unit and link service rates are (1, 0.5) packets per time unit.

(b) Average resequencing delay when fixed delays are (0, 0) time unit and link service rates are (1, 1) packet per time unit.

(c) Average resequencing delay when fixed delays are (0, 0) time unit and link service rates are (1, 2) packets per time unit.

(d) Average resequencing delay when fixed delays are (0, 0) time unit and link service rates are (1, 5) packets per time unit.

(e) Average resequencing delay when fixed delays are (0, 15) time units and link service rates are (1, 1) packet per time unit.

(f) Average resequencing delay when fixed delays are (15, 0) time units and link service rates are (1, 1) packet per time unit.
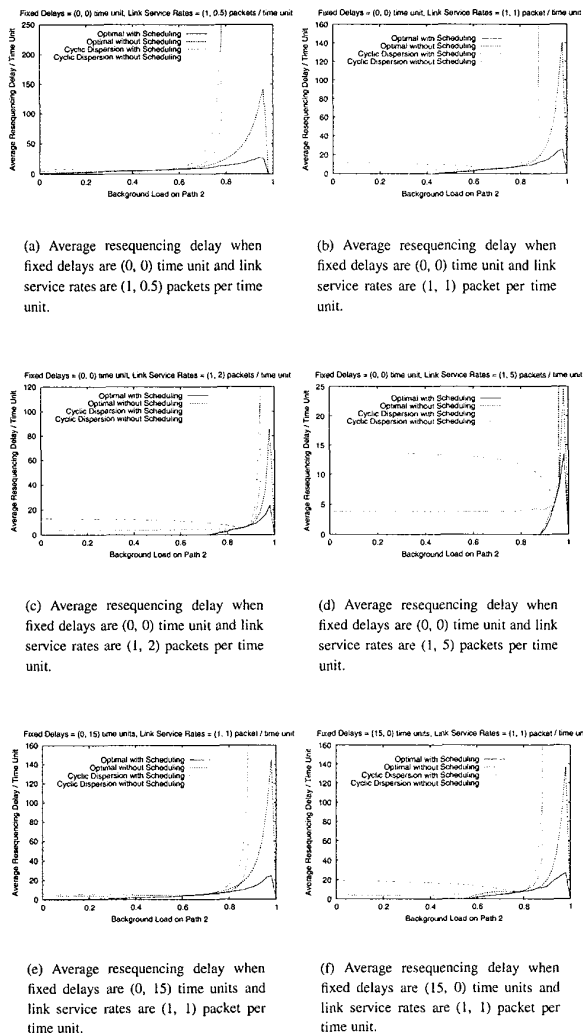
Fig. 4. Average resequencing delay plots for various settings.

packet. Yet, packets may still experience a substantial difference in average path delays if they are sent on different paths. The packet scheduling mechanism can be utilized to minimize or to reduce the consumption of the resequencing buffer and delay on resequencing by reducing the chance for out of order packet arrivals at the destination, say, by scheduling them according to their expected arrival times at the destination. This technique is particularly useful for transmitting archival information, such as stored videos, since all archival packets are available and they can be sent in any desired order.

The end-to-end path delay model with a delay line has been built to allow the queueing model for resequencing to be decoupled from that for a path. This leads to a simple yet general model, which can be used with other measurement-based tools for estimating end-to-end path delay distributions to find an optimal split of traffic.

To illustrate our proposed framework, we have considered a

multiple-node $M/M/1$ tandem network as a path model. When end-to-end path delays are Gaussian distributed, our analytical results show that the techniques are very effective in reducing the average end-to-end path delay, the average packet resequencing delay, and the average resequencing buffer occupancy for various path configurations. These promising results can form a basis for designing future adaptive multipath protocols.

There are several possible extensions to our work, some of which are listed below:

- devise an adaptive multipath protocols for packet-switching networks, such as IP-based and ATM networks;
- incorporate quality of service routing [5] with multipath routing;
- extend the framework to consider reliability and fault tolerant issues.

ACKNOWLEDGEMENT

REFERENCES

[1] Y. Arian and Y. Levy. Algorithms for Generalized Round Robin Routing. *Operations Research Letters*, Vol. 12, No. 5, pp. 313-319, November 1992.

[2] S. Bahk and M. E. Zarki. Preventive Congestion Control Based Routing in ATM Networks. *Proceedings of IEEE ICC '94*, Vol. 3, pp. 1592-1599, New Orleans, LA, USA, 1-5 May 1994.

[3] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger. Long-Range Dependence in Variable-Bit-Rate Video Traffic. *IEEE Transactions on Communications*, Vol. 43, No. 2-4, pp. 1566-1579, February-April 1995.

[4] D. Bertsekas and R. Gallager. *Data Networks*. Second Edition. Prentice Hall, 1992.

[5] S. Chen and K. Nahrstedt. An Overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions. *IEEE Networks*, Vol. 12, No. 6, pp. 64-79, November-December 1998.

[6] S. N. Chiou and V. O. K. Li. Diversity Transmissions in a Communication Network with Unreliable Components. *Proceedings of IEEE ICC '87*, Vol. 2, pp. 968-973, Seattle, WA, USA, 7-10 June 1987.

[7] J. H. Déjean, L. Dittmann, and C. N. Lorenzen. String Mode – A New Concept for Performance Improvement of ATM Networks. *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 9, pp. 1452-1460, December 1991.

[8] A. Erramilli, O. Narayan, and W. Willinger. Experimental Queueing Analysis with Long-Range Dependent Packet Traffic. *IEEE/ACM Transactions on Networking*, Vol. 4, No. 2, pp. 209-223, April 1996.

[9] E. Gustafsson. Traffic Dispersion in ATM Networks. Ph.D. Dissertation, Royal Institute of Technology, TRITA-IT R 97:03, Kista, Sweden, June 1997.

[10] W. W. Hines and D. C. Montgomery. *Probability and Statistics in Engineering and Management Science*. Third Edition. John Wiley & Sons, 1990.

[11] L. Kleinrock. *Queueing Systems (Volume I: Theory)*. John Wiley & Sons, 1975.

[12] L. Kleinrock. *Queueing Systems (Volume II: Computer Applications)*. John Wiley & Sons, 1976.

[13] R. Krishnan and J. A. Silvester. Choice of Allocation Granularity in Multipath Source Routing Schemes. *Proceedings of IEEE INFOCOM '93*, Vol. 1, pp. 322-329, San Francisco, CA, USA, 28 March - 1 April 1993.

[14] R. Krishnan and J. A. Silvester. An Approach to Path-Splitting in Multipath Networks. *Proceedings of IEEE ICC '93*, Vol. 3, pp. 1353-1357, Geneva, Switzerland, 23-26 May 1993.

[15] T. T. Lee, S. C. Liew, and Q.-L. Ding. Parallel Communications for ATM Network Control and Management. *Performance Evaluation*, Vol. 30, No. 4, pp. 243-264, October 1997.

[16] K.-C. Leung and V. O. K. Li. A Resequencing Model for High Speed Networks. *Proceedings of IEEE ICC '99*, Vol. 2, pp. 1239-1243, Vancouver, BC, Canada, 6-10 June 1999.

[17] V. O. K. Li and W. Liao. Distributed Multimedia Systems. *Proceedings of the IEEE*, Vol. 85, No. 7, pp. 1063-1108, July 1997.

[18] F. Y. S. Lin. Allocation of End-to-End Delay Objectives for Networks Supporting SMDS. *Proceedings of IEEE GLOBECOM '93*, Vol. 3, pp. 1346-1350, Houston, TX, USA, 29 November - 2 December 1993.

[19] D. G. Luenberger. *Linear and Nonlinear Programming*. Second Edition. Addison-Wesley, 1984.

[20] N. F. Maxemchuk. Dispersity Routing. *Proceedings of IEEE ICC '75*, pp. 41-10 - 41-13, San Francisco, CA, USA, June 1975.

[21] N. F. Maxemchuk. Dispersity Routing in High-Speed Networks. *Computer Networks and ISDN Systems*, Vol. 25, No. 6, pp. 645-661, January 1993.